

مهندسی نرم افزار مدل-رانده Model-Driven Software Engineering (MDSE)

دکتر بهمن زمانی

عضو هیأت علمی
گروه مهندسی کامپیوتر
دانشگاه اصفهان

ارائه شده در:
دانشگاه آزاد اسلامی - واحد شهر مجلسی

۱۰ فرورداد ۱۳۸۹

مفهوم مدل

➤ هر مدل نمایشی است از یک سامانه.

A model is a representation of a system

➤ **مدل مهندسی** یک سامانه، نمایشی از آن است که از میان ویژگی‌های آن سامانه، برخی را مخفی کرده و تنها آنهایی که مورد علاقه‌ی کاربر هستند را برجسته می‌کند.

A representation of a system that hides some of the properties and highlights the ones that are of interest for the user [Sel 06]

انواع مدل

۱. مدل ریاضی

■ مثال: برنامه ریزی خطی

Linear Programming

$$\begin{array}{ll} \text{minimize} & \sum_{j=1}^n c_j x_j \\ \text{subject to} & \sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = 1, 2, \dots, m) \\ & x_j \geq 0 \quad (j = 1, 2, \dots, n) \end{array}$$

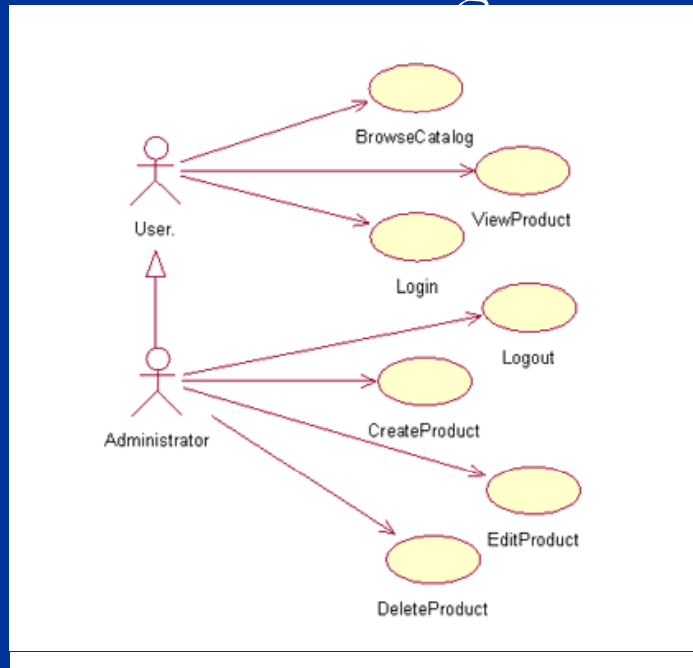
<http://www.cise.ufl.edu/~davis/Morgan/Image13.gif>

انواع مدل

۲. مدل نموداری

■ مثال: مدل سازی موارد استفاده

Use Case Modeling



<http://www.javaworld.com/javaworld/jw-07-2004/images/jw-0719-jsf1.gif>

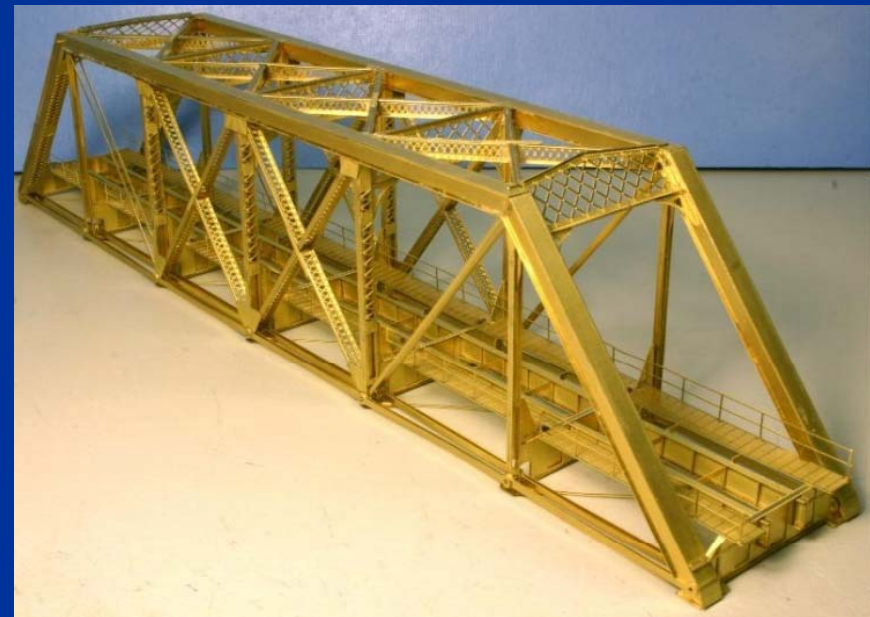
انواع مدل

۳. مدل فیزیکی

■ مثال: مدل سازی در صنایع



<http://yourmilitaryaircraft.com/pics/c130-20.jpg>



<http://texnrails.com/images/track/tms/1200701.jpg>

انواع مدل

۳. مدل فیزیکی

■ مثال: مدل سازی در ساختمان



<http://www.isfahancitycenter.com/soffe6.jpg>

چرا مدل؟

- در اغلب رشته‌های مهندسی، رسم بر این است که برای طراحی یک سامانه‌ی پیچیده از مدل استفاده کنند. [Sel 03]
- از آنجا که سامانه‌های نرم‌افزاری جدید نیز رفته رفته پیچیده می‌شوند، ما **(نرم‌افزاری‌ها)** هم ناگزیر به استفاده از مدل هستیم. [Sel 03]

مدل سازی و مهندسی نرم افزار

➤ در آخرین ویرایش (چاپ 2010) کتاب مهندسی نرم افزار Pressman تمامی مراحل تحلیل و طراحی نرم افزار در بخشی با عنوان «مدل سازی» قرار گرفته است.

Part Two Modeling

4 Principles that Guide Practice

5 Understanding Requirements

6 Requirements Modeling: Scenarios, Information, and Analysis Classes

7 Requirements Modeling: Flow, Behavior, Patterns, and WebApps

8 Design Concepts

9 Architectural Design

10 Component-Level Design

11 User Interface Design

12 Pattern-Based Design

13 WebApp Design

[Pre 10]

روش‌های مدل-رانده در توسعه نرم‌افزار

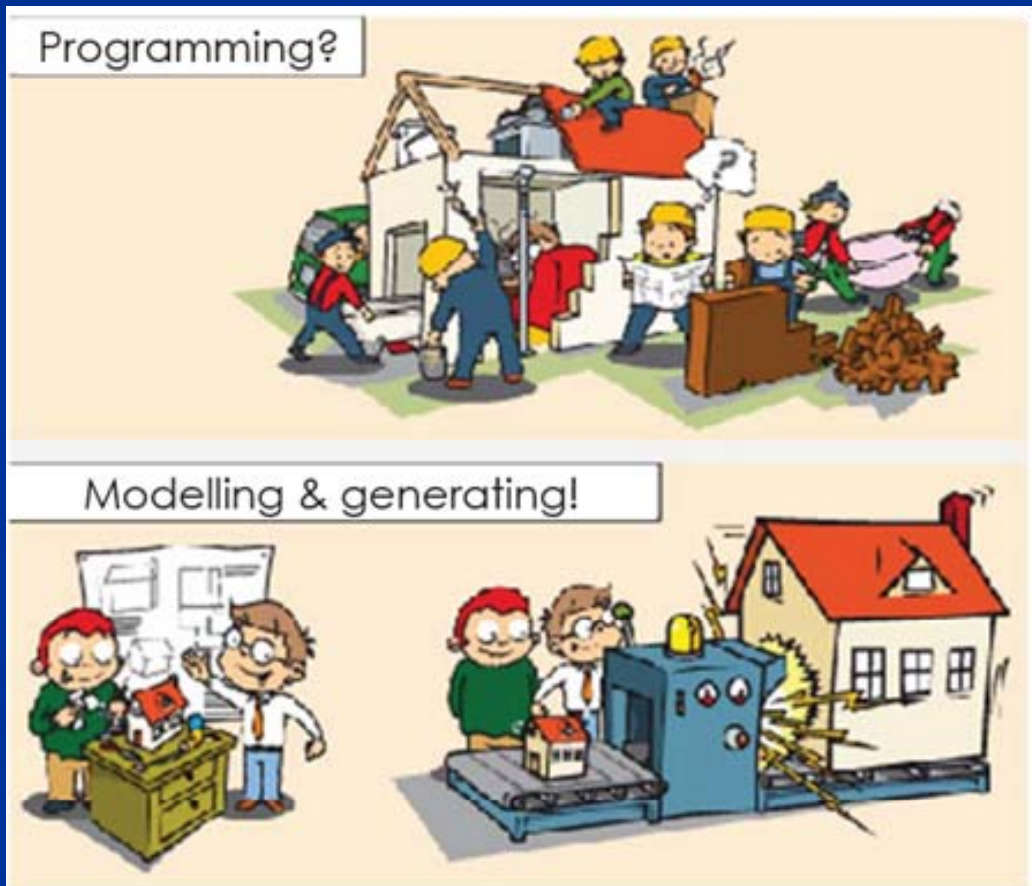
- اغلب فرآیندهای توسعه نرم‌افزار کد-محور هستند.
- ولی در روش‌های مدل-رانده، مدل محصول اصلی است و هدایت توسعه نرم‌افزار را بر عهده دارد.
- این مدل است که می‌راند ← **مدل-رانده**



<http://www.clker.com/clipart-2403.html>

روش‌های مدل-رانده در توسعه نرم‌افزار

➤ هدف نهایی این است که کد را به طور خودکار از روی مدل
به دست آوریم. [Sel 03]



<http://www.theenterprisearchitect.eu/archive/2009/08/05/a-metaphor-for-model-driven-engineering>

انواع روش‌های مدل-رانده

➤ معماری مدل-رانده

MDA: Model-Driven Architecture

➤ توسعه‌ی مدل-رانده

MDD: Model-Driven Development

➤ مهندسی مدل-رانده

MDE: Model-Driven Engineering

➤ توسعه‌ی نرم‌افزار به روش مدل-رانده

MDSD: Model-Driven Software Development

➤ مهندسی نرم‌افزار به روش مدل-رانده

MDSE: Model-Driven Software Engineering

کدام روش مدل-رانده؟

MDA, MDD, MDE, MDSD, MDSE?

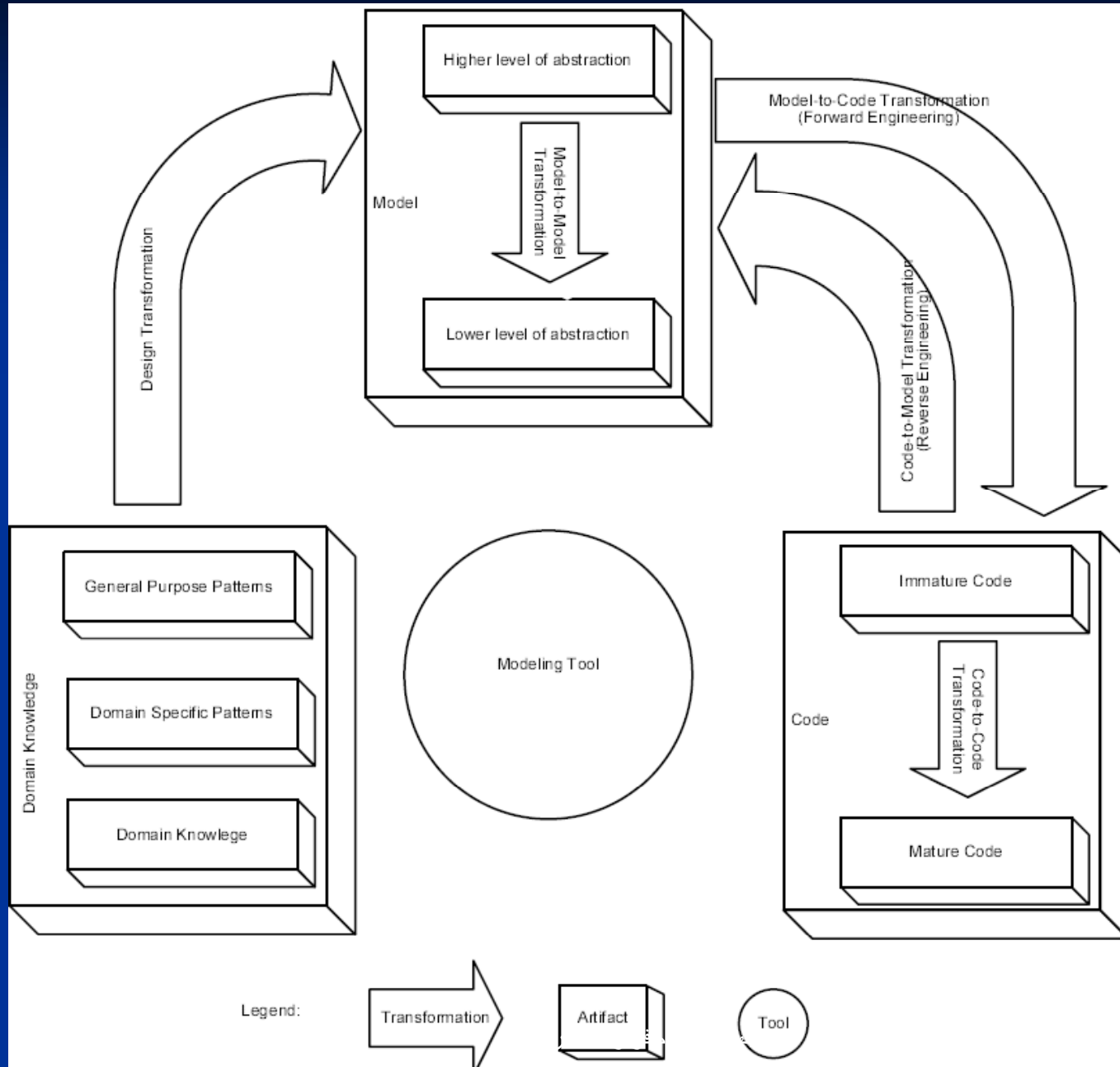


http://wwwt-one.net/~om/NCTblog/ist2_2771207_dizzy.jpg

Don't worry, just go **Model-Driven!**

➤ نگران اصطلاحات نباشید، فقط اهمیت مدل و مدل‌سازی را فراموش نکنید.

یک نقشه‌ی راه برای مهندسی مدل-رانده



Source: My PhD Thesis

زبان مدل سازی یکنواخت

The Unified Modeling Language (UML)



<http://www.uml.org/>

➤ یک زبان گرافیکی برای

■ تبیین (specifying)

■ مصورسازی (visualizing)

■ ساخت (constructing)

■ و مستندسازی (documenting) محصولات نرم افزاری

➤ روش غالب برای مدل سازی گرافیکی سامانه های نرم افزاری شیء گرا

➤ نسخه های UML

- UML 1.1 adopted by OMG in November 1997
- Current release is UML 2.3 (released 9/9/09).

نمودارهای UML 2.0

[<http://www.agilemodeling.com/essays/umlDiagrams.htm>]

1. Activity Diagram
2. Class Diagram
3. Communication Diagram
4. Component Diagram
5. Composite Structure Diagram
6. Deployment Diagram
7. Interaction Overview Diagram
8. Object Diagram
9. Package Diagram
10. Sequence Diagram
11. State Machine Diagram
12. Timing Diagram
13. Use Case Diagram

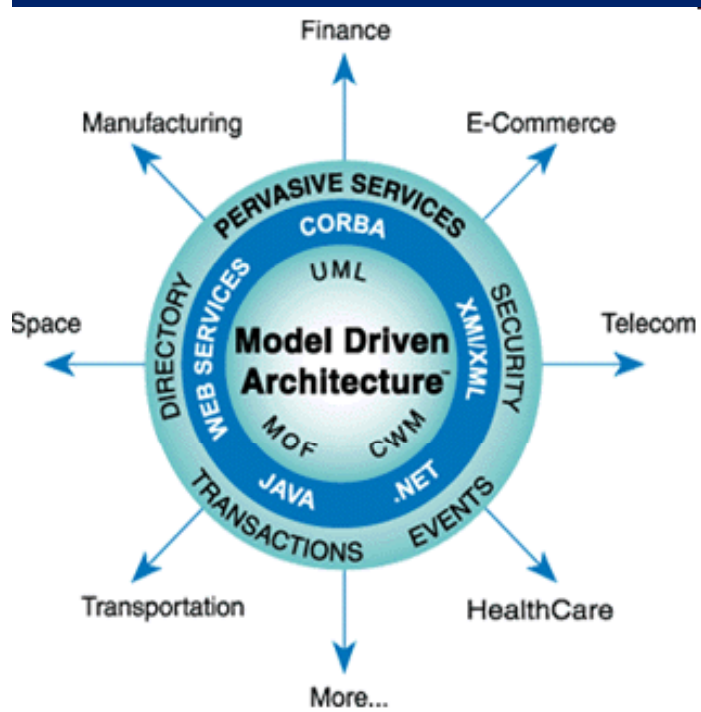
آیا واقعاً روش مدل-رانده کار می‌کند؟

➤ دو راه حل (تقریباً) عملی

■ معماری مدل-رانده (MDA)

■ UML اجرایی (xUML)

Model-Driven Architecture (MDA)



<http://www.omg.org/mda/>

➤ دو چیز را از هم جدا می کند

■ تبیین عمل کرد (آنچه که قرار است سامانه انجام دهد)

■ تبیین پیاده سازی (جزئیات پیاده سازی)

➤ با دو نوع مدل سر و کار دارد

■ مدل مستقل از سگو

Platform-Independent Model (PIM)

■ مدل وابسته به سگو

Platform-Specific Model (PSM)

➤ شعار: طراحی یک بار، ساخت روی هر سگو

Design once, build it on any platform (.NET, J2EE, CORBA,...)

مدل مستقل از سکو (PIM)

- نمایش صوری و مجردی از ساختار و عملکرد سامانه، بدون پرداختن به جزئیات فنی

An abstract “formal” specification of the structure and function of a system, leaving out the technical details

- که به وسیلهی UML نمایش داده می‌شود

- هر جنبه‌ای (یا نگرشی) از سامانه توسط یک PIM نمایش داده می‌شود.

Each PIM captures an aspect (or viewpoint) of the system

- قلمرو هم نامیده می‌شوند.
- سامانه‌ی مورد نظر از ترکیب قلمروها به دست می‌آید.

مدل وابسته به سگو (PSM)

- تمامی کاربردهای معین شده در مدل مستقل از سگو را در بر می‌گیرد، به اضافه‌ی نکات طراحی آنها.

The PSM comprises all the functionality expressed in the PIM with the added design concerns

- به وسیله‌ی UML بیان می‌شود، منتهی جزئیات مختص سگو نیز به آن اضافه شده است.

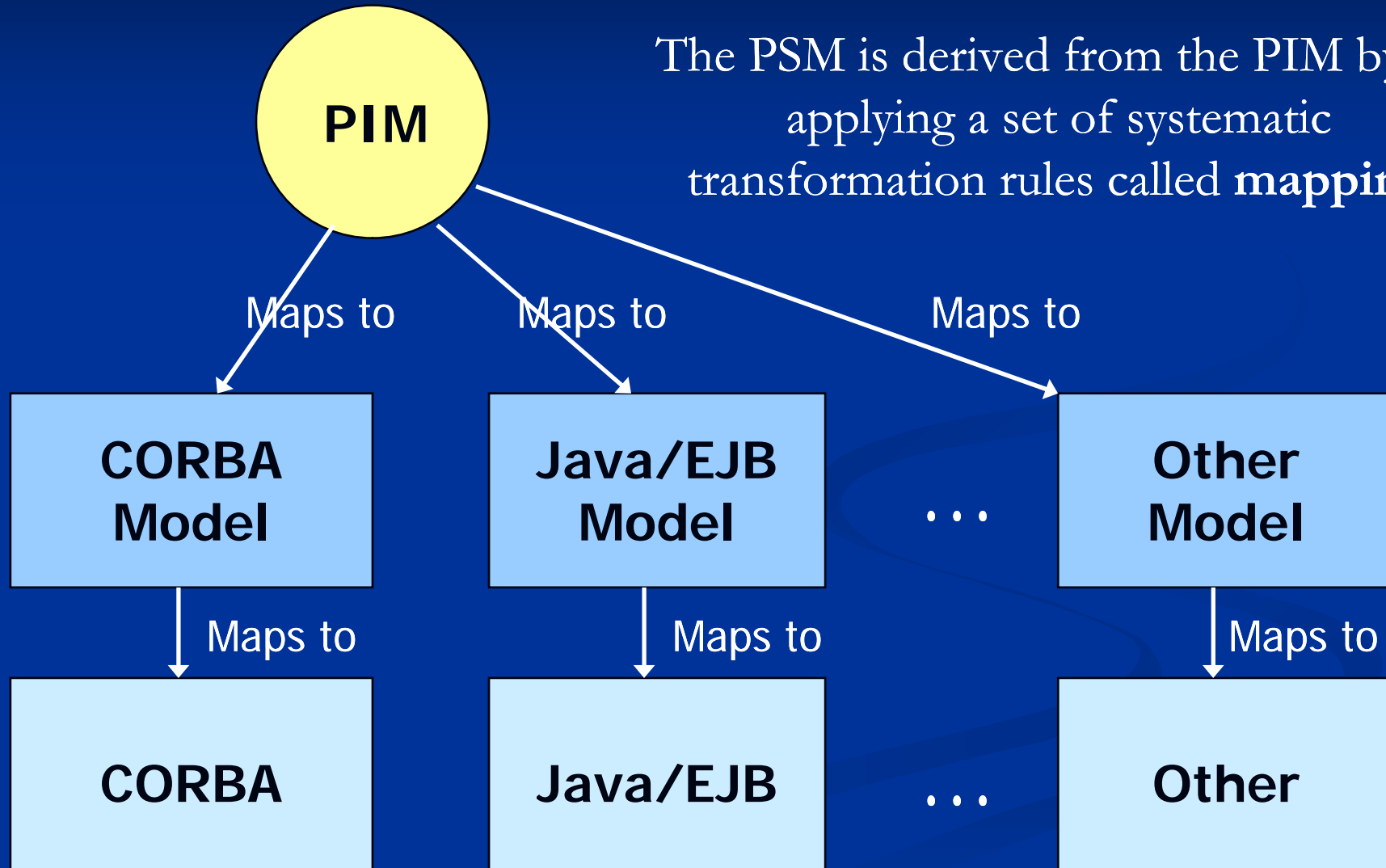
PSM is expressed using UML extended by adding platform specific details

- با استفاده از پروفایل‌ها

Using UML extension mechanisms (UML Profiles)

نگاشت از PIM به PSM

The PSM is derived from the PIM by applying a set of systematic transformation rules called **mapping**



Executable UML (xUML)

- نسخه‌ی قابل اجرایی از UML
- یک متدلوژی خودکار بر اساس زیرمجموعه‌ای کاملاً تخصصی شده از نمادهای UML مانند

UML Class Diagrams, State Charts, Action Language

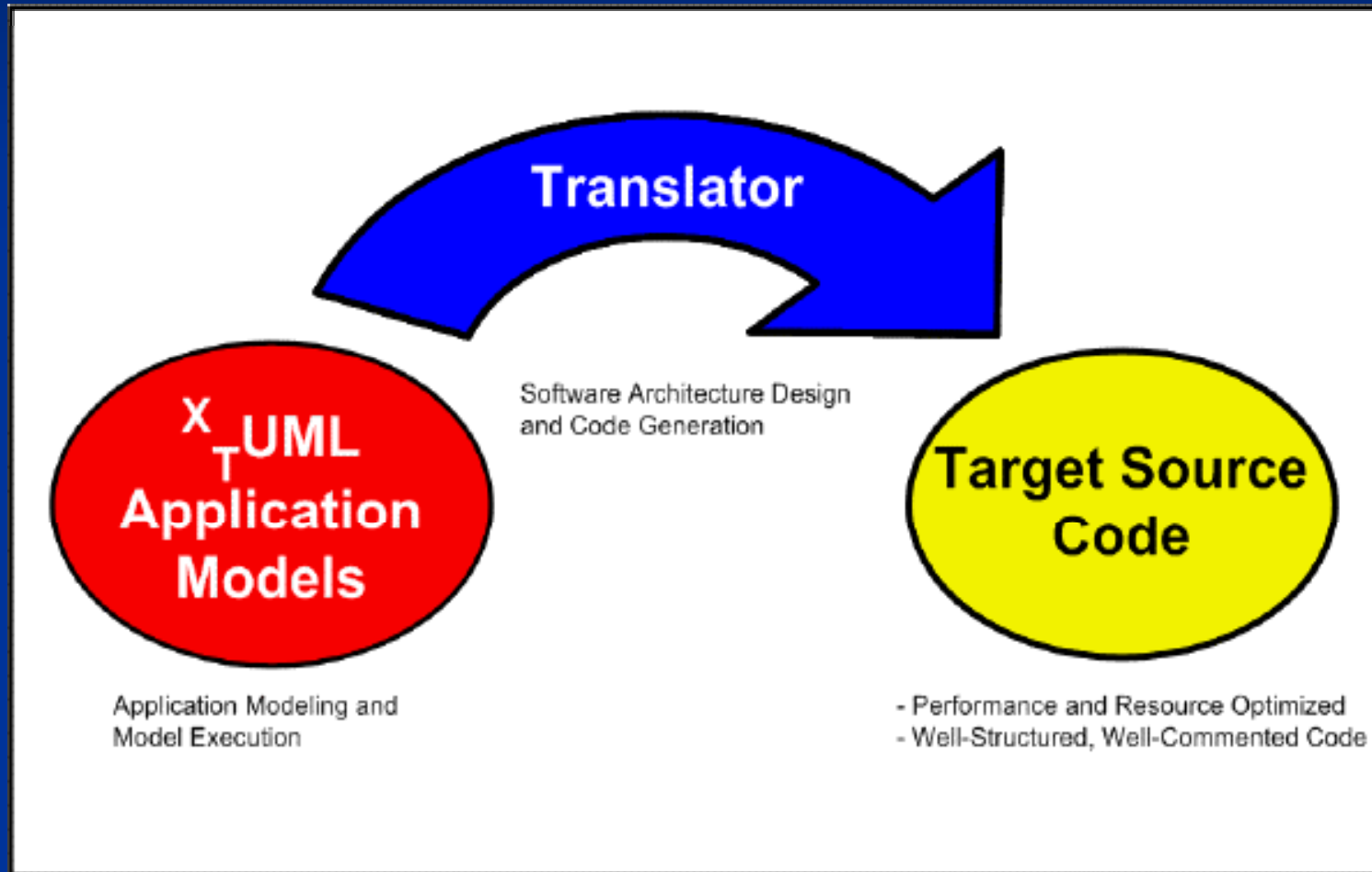
xUML →

UML

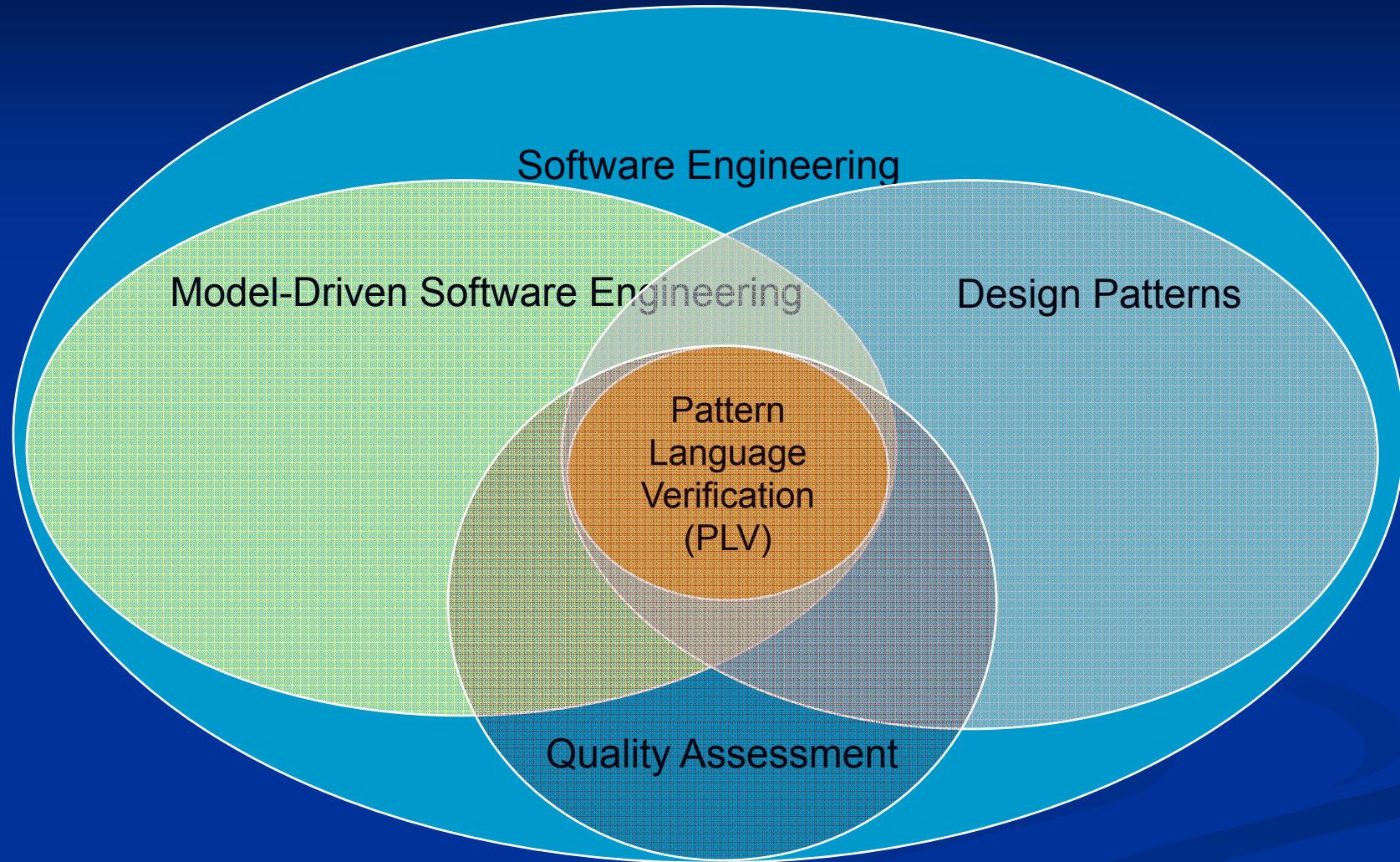
Semantically weak elements

Action Semantic

xUML tool: BridgePoint Development Suite



زمینه‌ی تحقیقاتی



مراجع

- [Sel 03] Bran Selic. The pragmatics of model-driven development. *IEEE Software*, 20(5):19–25, Sep. 2003.
- [Sel 06] Bran Selic. Model-driven development: Its essence and opportunities. In *Proc. ISORC'06*, pages 313–319, Los Alamitos, CA, USA, 2006. IEEE Computer Society Press.
- [Obj 05] Object Management Group (OMG). Unified Modeling Language (UML): Infrastructure, v2.0. OMG document: formal/05-07-05, 2005.
- [Pre 10] Roger S. Pressman, *Software Engineering: A Practitioner's Approach*, Seventh Edition, McGraw-Hill, 2010.

متشکرم

- برای اطلاعات بیشتر به وبگاه من مراجعه کنید
- Visit my page at bahmazamani.com



<http://www.greateracadianaregion.net/edu/Portals/0/images/cct/Questions.jpg>